



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Am

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/801,150	03/07/2001	Thierry Bedos	10984.3US01	4307

25883 7590 03/09/2005
HOWISON & ARNOTT, L.L.P
P.O. BOX 741715
DALLAS, TX 75374-1715

EXAMINER

YIGDALL, MICHAEL J

ART UNIT PAPER NUMBER

2122

DATE MAILED: 03/09/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/801,150

Applicant(s)

BEDOS ET AL.

Examiner

Michael J. Yigdall

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 24 November 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-3, 7-20, 22 and 25-39 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-3, 7-20, 22 and 25-39 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on November 24, 2004 has been entered. Claims 1-3, 7-20, 22 and 25-39 remain pending.

Response to Arguments

2. Applicant's arguments have been fully considered but they are not persuasive.

3. Applicant contends that the cited sections of Hawkins do not describe a non-core service as providing functionality that corresponds to user interaction with the application during run-time, and likewise that Hawkins does not describe analyzing user interaction with a program during run-time (Applicant's remarks, page 7, second paragraph).

However, the class files disclosed by Hawkins comprise non-core services that provide functionality that corresponds to user interaction with the application during run-time (see, for example, column 1, lines 29-34, which shows that "the class files to be downloaded depend on the objects instantiated during applet program execution," and column 4, lines 23-40, which further shows that the functionality provided by the class files corresponds to user interaction with the application during run-time). A user supplying input to the application, such as described by Hawkins (column 4, lines 23-39), is an example of user interaction with the application during run-time.

Moreover, Hawkins expressly discloses analyzing user interaction with a program during run-time (see, for example, column 3, lines 57-65). Applicant acknowledges that Hawkins discloses an analysis based on tracing (Applicant's remarks, page 7, second paragraph), and Hawkins further discloses that this tracing involves test cases "where a user runs the application" (column 3, lines 57-65). Tracing must be performed during run-time; if the application were not running, there would be nothing to trace.

4. Applicant contends that the cited sections of Hawkins do not describe determining the order in which classes or services are required for providing functionality corresponding to user interaction, and likewise that in Hawkins, user interaction has no effect on the downloading of classes during run-time (Applicant's remarks, page 8, first paragraph).

However, Hawkins expressly discloses determining the order in which classes or services are required for providing functionality that corresponds to user interaction (see, for example, column 4, lines 10-39, which shows determining the order in which objects are instantiated from the classes according to user interaction). Furthermore, as discussed above, Hawkins discloses analyzing user interaction with the program during run-time (see, for example, column 3, lines 57-65). The purpose of this analysis is to determine how the classes are to be downloaded during run-time (see, for example, column 3, lines 9-26). Therefore, in Hawkins, user interaction affects the downloading of classes during run-time.

5. Applicant contends that the cited sections of Hawkins do not describe designating non-core services as a top priority in the priority order in response to user interaction during run-time, and likewise that in Hawkins, there is no designation of a particular non-core service being

designated as a top priority in the priority order such that the functionality of the application is enabled in response to user interaction during run-time (Applicant's remarks, page 8, second paragraph to page 9, top).

However, as discussed above, Hawkins discloses determining the order in which the non-core services provided by the class files are used during run-time (see, for example, column 4, lines 10-39). Hawkins further discloses designating a load sequence or a priority order for the classes (see, for example, FIG. 3 and column 5, lines 17-32). The priority order specifies when the classes are to be downloaded so as to enable functionality of the application (see, for example, column 6, lines 26-35). Again, the analysis that provides the priority order is based on user interaction during run-time (see, for example, column 3, lines 57-65). Classes that are required before other classes inherently have a higher priority (see, for example, column 6, lines 4-8, which shows downloading the top-priority classes before other classes).

6. In response to Applicant's characterization of Hawkins (Applicant's remarks, page 9-10) and Applicant's assertion that the present invention provides "a much more robust and dynamic operating environment" (Applicant's remarks, page 9, second paragraph), it should be noted that the language of the claims does not positively exclude Hawkins' disclosure. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

Art Unit: 2122

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

8. Claims 1-3, 7, 8, 16-20, 22 and 32 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Pat. No. 6,536,035 to Hawkins (art of record, “Hawkins”).

With respect to claim 1 (previously presented), Hawkins discloses a software engine for application loading a software application onto a user’s machine (see, for example, the title), wherein a core service of the application is loaded onto the user’s machine to enable the application to commence to operate on the user’s machine (see, for example, FIG. 1 and column 4, lines 10-12, which shows loading core classes or services to commence the operation of a program), the engine subsequently loading non-core services of the application according to a priority order determined by the engine (see, for example, column 3, lines 22-26, which shows a software engine for preloading class files of non-core services during program execution, and column 5, lines 57-67, which shows that the class files of non-core services are loaded in the correct order using a time-based priority order), wherein a non-core service is responsible for providing a functionality of the application and corresponds to a user interaction with the application during run-time (see, for example, column 3, lines 57-65, which shows analyzing user interaction with the program at run-time, and column 4, lines 10-39, which shows determining the order in which classes or services are required for providing functionality corresponding to user interaction), and wherein, in response to the user interaction during run-time, corresponding non-core services are designated a top priority in the priority order such that

Art Unit: 2122

functionality of the application is enabled (see, for example, FIG. 3 and column 5, lines 17-32, which shows examples of the time-based priority order determined by user interaction; note that top priority is inherently assigned to the classes or services required before others).

With respect to claim 2 (original), Hawkins also discloses the limitation wherein the engine is part of the core service and is loaded with the core service (see, for example, column 5, lines 57-67, which shows adding the engine to the first portion of the program, i.e. so that it will be loaded with the core services).

With respect to claim 3 (original), Hawkins also discloses the limitation wherein the engine commences operation upon completion of loading of the core service (see, for example, column 6, lines 4-8, which shows commencing operation of the engine immediately after the core services are loaded and instantiated).

With respect to claim 7 (previously presented), Hawkins also discloses the limitation wherein before loading the non-core services they are registered with the engine (see, for example, column 3, line 66 to column 4, line 9, which shows determining whether classes or services have been previously referenced; note that in order to make this determination, the classes or services are inherently registered with the engine when they are first instantiated).

With respect to claim 8 (previously presented), Hawkins also discloses the limitation wherein the engine checks a registration list of non-core services before loading a requested non-core service (see, for example, column 3, line 66 to column 4, line 9, which shows determining

Art Unit: 2122

whether classes or services have been previously referenced, i.e. checking a registration list, prior to loading them).

With respect to claim 16 (original), Hawkins also discloses the limitation wherein the loading is downloading over the Internet (see, for example, column 3, lines 9-21, which shows downloading an Internet application).

With respect to claim 17 (previously presented), the limitations recited in the claim are analogous to those of claim 1 (see the rejection of claim 1 above).

With respect to claim 18 (original), the limitations recited in the claim are analogous to those of claim 2 (see the rejection of claim 2 above).

With respect to claim 19 (original), the limitations recited in the claim are analogous to those of claim 3 (see the rejection of claim 3 above).

With respect to claim 20 (original), the limitations recited in the claim are analogous to those of claim 7 (see the rejection of claim 7 above).

With respect to claim 22 (previously presented), Hawkins also discloses the limitation wherein upon interaction with the application by the user, the application requests the engine to load at least one of the non-core services (see, for example, column 4, lines 23-39, which shows loading non-core classes or services based on user interaction), and the engine checks a registration and gives the at least one non-core service top priority for loading (see, for example, column 3, line 66 to column 4, line 9, which shows determining whether classes or services have

Art Unit: 2122

been previously referenced, i.e. checking the registration list; also see, for example, column 5, lines 57-67, which shows that the classes or services are loaded using a time-based priority order; note that top priority is inherently assigned to the class or service required first).

With respect to claim 32 (original), the limitations recited in the claim are analogous to those of claim 16 (see the rejection of claim 16 above).

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. Claims 9-15, 25-31 and 33-39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hawkins, as applied to claims 1 and 17 above, respectively, in view of U.S. Pat. No. 6,430,570 to Judge et al. (art of record, "Judge").

With respect to claim 9 (original), although Hawkins discloses that class files may be cached on a client machine prior to execution (see, for example, column 1, lines 34-40), Hawkins does not expressly disclose the limitation wherein there is provided a cache into which at least one object for the application can be stored.

However, Judge discloses the limitation above in terms of storing class objects in a cache to reduce the number of redundant downloads, thereby improving performance (see, for example, application cache 52 in FIG. 2 and column 7, lines 28-36).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 10 (original), Hawkins does not expressly disclose the limitation wherein the engine includes a memory management module that keeps track of usage of cached objects; the memory management module being able to de-allocate one or more of the objects.

However, Judge further discloses the limitation above in terms of an application memory manager that keeps track of cache objects and removes objects for garbage collection in order to free up space in memory (see, for example, column 8, lines 37-47).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 11 (original), Hawkins also discloses the limitation wherein the cache is operative only when the application is on the user's machine (see, for example, column 1, lines 34-40, which shows that class files may be cached on a client machine prior to execution).

With respect to claim 12 (original), Hawkins does not expressly disclose the limitation wherein the cache includes an object repository into which the at least one object is placed, and an object description.

However, Judge further discloses the limitation above in terms of caching class objects in a repository along with a description comprising an object reference (see, for example, application cache 52 in FIG. 2 and column 7, lines 45-51).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 13 (original), Hawkins does not expressly disclose the limitation wherein the object description includes one or more selected from the group consisting of: object reference, object key, reference counter and time stamp.

However, Judge further discloses the limitation above in terms of maintaining an object reference (see, for example, column 7, lines 45-51).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 14 (original), Hawkins does not expressly disclose the limitation wherein the de-allocation of one or more of the objects includes an arbitrary time offset.

However, Judge further discloses the limitation above in terms of continuously monitoring the free memory level and removing objects for de-allocation by the garbage collector as needed (see, for example, column 8, lines 43-47; note that continuously monitoring the free memory level inherently involves polling at an arbitrary time interval or offset).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 15 (original), Hawkins does not expressly disclose the limitation wherein if the object description of an object in the object repository has a reference counter equal to zero for a time equal to at least the time offset, the corresponding object description is removed from the object repository.

However, Judge further discloses the limitation above in terms of maintaining an object reference to each class to protect it from garbage collection (see, for example, column 7, lines 45-51), meaning that the objects will be removed when the reference counter is equal to zero. Judge further discloses that the garbage collection will take place when a low or no free memory level is detected, inherently after an arbitrary time offset due to polling (see, for example, column 8, lines 43-47).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 25 (original), the limitations recited in the claim are analogous to those of claim 9 (see the rejection of claim 9 above). Note that Judge further discloses that the class objects are stored in the cache for later reuse (see, for example, column 7, lines 45-51).

With respect to claim 26 (original), the limitations recited in the claim are analogous to those of claim 10 (see the rejection of claim 10 above).

With respect to claim 27 (original), the limitations recited in the claim are analogous to those of claim 11 (see the rejection of claim 11 above).

With respect to claim 28 (original), the limitations recited in the claim are analogous to those of claim 12 (see the rejection of claim 12 above).

With respect to claim 29 (original), the limitations recited in the claim are analogous to those of claim 13 (see the rejection of claim 13 above).

With respect to claim 30 (original), the limitations recited in the claim are analogous to those of claim 14 (see the rejection of claim 14 above).

With respect to claim 31 (original), the limitations recited in the claim are analogous to those of claim 15 (see the rejection of claim 15 above).

With respect to claim 33 (previously presented), although Hawkins discloses that class files may be cached on a client machine prior to execution (see, for example, column 1, lines 34-40), Hawkins does not expressly disclose a computer memory management system including a cache, and wherein objects of the application are storable in the cache for reuse.

However, Judge discloses the limitation above in terms of an application memory manager (see, for example, column 8, lines 37-42) that includes a cache for storing class objects (see, for example, application cache 52 in FIG. 2 and column 7, lines 28-36), so that the objects may be reused (see, for example, column 7, lines 45-51).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the caching features taught by Judge in the system of Hawkins, for the purpose of improving performance (see, for example, Judge, column 7, lines 28-29).

With respect to claim 34 (previously presented), the limitations recited in the claim are analogous to those of claim 11 (see the rejection of claim 11 above).

With respect to claim 35 (previously presented), the limitations recited in the claim are analogous to those of claim 12 (see the rejection of claim 12 above).

With respect to claim 36 (previously presented), the limitations recited in the claim are analogous to those of claim 13 (see the rejection of claim 13 above).

With respect to claim 37 (previously presented), the limitations recited in the claim are analogous to those of claim 10 (see the rejection of claim 10 above).

With respect to claim 38 (previously presented), the limitations recited in the claim are analogous to those of claim 14 (see the rejection of claim 14 above).

With respect to claim 39 (previously presented), the limitations recited in the claim are analogous to those of claim 15 (see the rejection of claim 15 above).

Conclusion

11. The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure. U.S. Pat. No. 6,807,559 to Budhiraja discloses a system and method for applet

management. U.S. Pat. No. 6,279,030 to Britton et al. discloses dynamic Java class selection and download based on changeable attributes.

12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (571) 272-3707. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

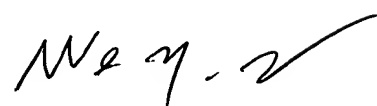
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MY

Michael J. Yigdall
Examiner
Art Unit 2122

mjy


WEI Y. ZHEN
PRIMARY EXAMINER